

Pixel Calibration Status

Pixel Database Meeting, 9/14/2006

- Introduction
- Data Model
- Implementation
- What Next



Introduction

- The calibration data is stored in the database with defined format and its Interval of Validity (IOV)
- A tool/service is used to access the data via COOL API
- Data consists of:
 - Threshold, sigma, noise
 - Time walk or in-time threshold
 - Time over Threshold conversion ($ToT = A + B/(Q+C)$, $\sigma=P1+P2 \cdot ToT$)
- Payload is BLOB containing constants for each chip
- Extracting the on-line averages for standard, long, ganged pixel per chip
- Defining Objects in Transient Detector Store (TDS) and Interfacing with offline packages
- Follow some of examples in SCT and TRT calibration.

Data Model

Storage	Definition	Units	Typical range
1B	threshold	100e	2000-5000 e
1B	dispersion	10e	80 - 600 e
1B	noise	10e	0 - 600 e
1B	timewalk	100e	2000-8000 e
float	A for ToT		0-300
float	B for ToT		
float	C for ToT		
1B	P1 dispersion of ToT	1/100	0-40
1B	P2 dispersion of ToT	1/1000	0-100

- 38B/per chip, corresponding to 1 MB/Detector
- It's small size, easy to handle ?

Object Class in TDS

- Create PixelChipSummaryData to hold all the data for a chip
 - chipID m_ident; // std::map<Wafer_id, n(0-15) >
 - char m_threshold[4]; // (thres, sigma, Noise,TW):
 - char m_thresholdlong[4]; // (thres, sigma, Noise,TW) for long pixel
 - char m_thresholdganged[4]; // (thres, sigma, Noise,TW) for ganged pixel
 - float m_q2tot[3]; // a,b,c for standard pixel and long
 - float m_q2totganged[3]; // a,b,c for ganged
 - char m_totres[2]; // for (p1, p2) each chip
- Store PixelCalibData in single folder '/Pixel/Calib/' or multiple folders based on barrel and layers.
 - std::string m_tag
 - std::vector<PixelChipSummaryData> m_vec

PixelCalibDataColl

- `typedef CondMultiChanCollection<PixelCalibData> PixelCalibDataColl`
- Saved to storegate and persistified using POOL
- Accessories:
 - `const chipID& getChipID() const;`
 - `int getThreshold(int type); //type=0/1/2 for standard,long and ganged pixels`
 - `int getThresholdSigma(int type), getNoise(int type), getTimeWalk(int type)`
 - `float getQ2TotA(int type), getQ2TotB(int type), getQ2TotC(int type);`
 - `float getTot(float Q), getTotP1(), getTotP2(), getTotRes(float Q);`
 - `void setChipID(const chipID& ident);`
 - `void setThreshold(int type, int thres, int sigma, int noise, int timewalk);`
 - `void setQ2Tot(int type, float A, float B, float C);`
 - `void setTotRes(int p1, int p2);`

PixelCalibDbTool

- All access is provided in the AlgTool – PixelCalibDbTool
- Call a get/set method with the name of the desired quantity passing the identifier of detector elements.
- Methods are also provided for bulk transfer of constants between text file and IOV/POOL files.
- Abilities to stream data to POOL file and register them in the IOV database
- Providing test algorithms to read and write to database.

What Next

- Continue prototyping and improving
- Provide PixelCalibDbSvc Db service
- Commit the code to repository
- Preparing Pixel Calibration data from CTB and production data
- Interface with offline packages
- Timing and performance studies
- ...

